# CAFE MANAGEMENT SYSTEM

A Project Submitted to Madurai Kamaraj University in Partial Fulfilment

Of the Requirement for the Award of Degree of

**BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY**

**Submitted by**

2020IT17 MEKALA. S

Under the Guidance of

Dr. V. Jane Varamani Sulekha, M.Sc., MCA, M.Phil., SET., Ph.D., NET

Assistant Professor

Department of Information Technology

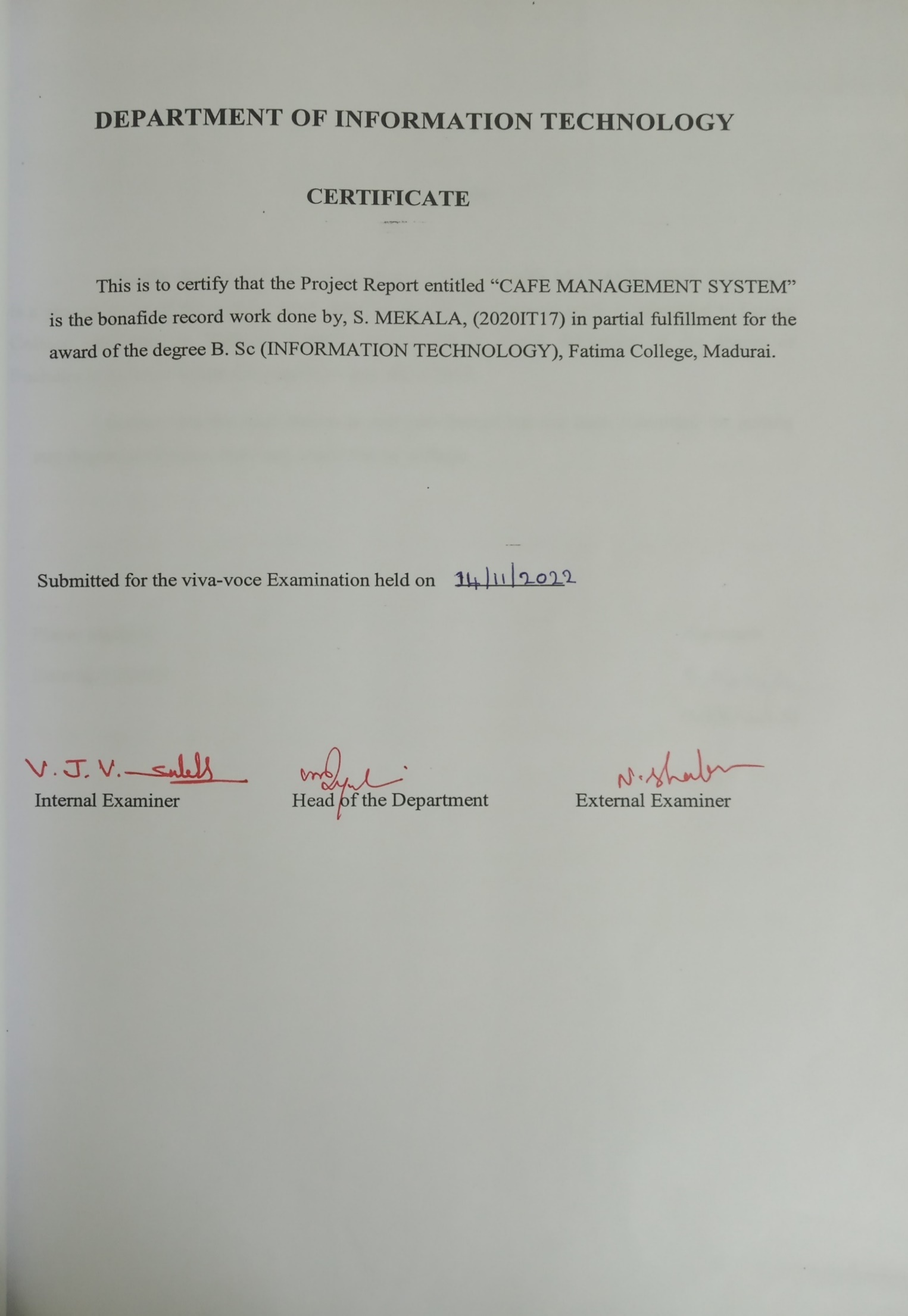


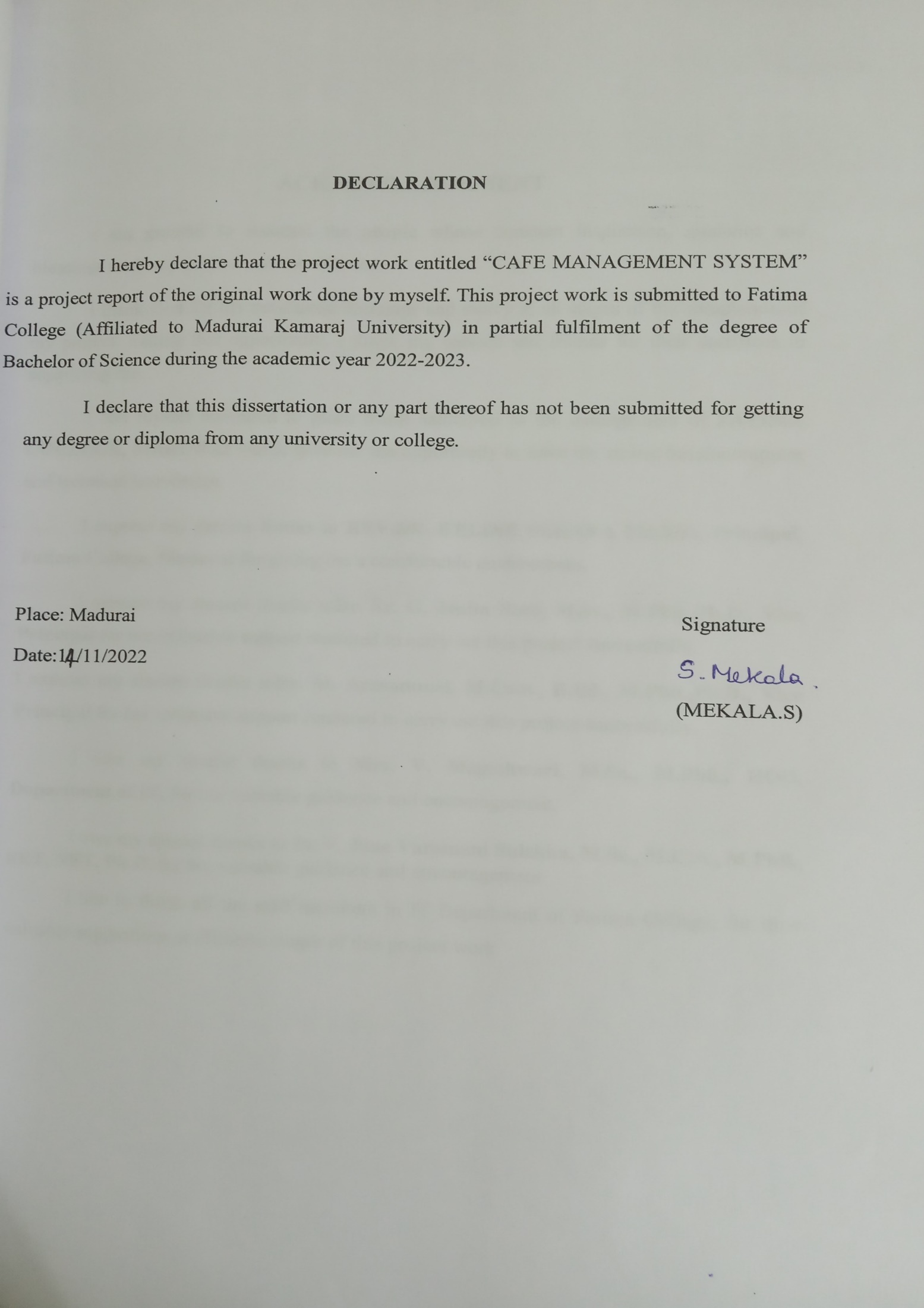
FATIMA COLLEGE (AUTONOMOUS)

Re-Accredited with ‘A++’ Grade by NAAC (4th Cycle)

Mary Land, Madurai - 625018, Tamil Nadu.

**NOVEMBER - 2022**





**ACKNOWLEDGEMENT**

I am grateful to mention the people whose constant inspiration, guidance and blessings made my project a successful one.

I thank God for his substantial blessings and mercy at all stages of the completion of my project. Taking this opportunity, I thank my parents and friends for their sacrifices in supporting me.

I feel myself honoured to place warm salutation to the management of **FATIMA COLLEGE, MADURAI** which gave me the opportunity to have the strong baseincomputer and technical knowledge.

I express my sincere thanks to **REV. DR. CELINE SAHAYA MARY., Principal, Fatima College, Madurai** for giving me a comfortable environment**.**

I express my sincere thanks to **Dr. Sr. G. Jenita Rani, M.Sc., M.Phil., Ph.D., Vice Principal** for her extensive support rendered to carry out this project successfully.

I express my sincere thanks to **Dr. M. Arasammal, M.Com., B.Ed., M.Phil., Ph.D., Vice Principal** for her extensive support rendered to carry out this project successfully.

I owe my special thanks to **Mrs. V. Mageshwari, M.Sc., M.Phil., HOD, Department of IT,** for her valuable guidance and encouragement.

I owe my special thanks to **Dr.** **V. Jane Varamani Sulekha, M.Sc., M.C.A., M.Phil., SET, NET, Ph.D.** for her valuable guidance and encouragement

I like to thank all the staff members in IT Department of Fatima College, for their valuable suggestions at different stages of this project work

**CONTENT**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **Title** | **Page No** |
| 1 | **1. INTRODUCTION**  **1.1 Abstract** | 1  2 |
| 2 | **2. SYSTEM ANALYSIS**  2.1 Existing System  2.2 Proposed System  2.3 Module Description | 3  4  5 |
| 3 | **3. SYSTEM SPECIFICATION**  3.1 Hardware Requirement  3.2 Software Requirement  3.3 Software Description | 7  8 |
| 4 | **4. SYSTEM DESIGN**  4.1 Data Flow Diagram  4.2 Entity Relation Diagram  4.3 Table Structure | 22  24  25 |
| 5 | **5. PROJECT IMPLEMENTATION**  5.1 Sample Coding  5.2 Sample Output | 28  38 |
| 6 | **6. TESTING**  6.1 Unit Testing | 46  47 |
| 7 | **7. CONCLUSION**  7.1 Benefits  7.2 Future Enhancement | 48  49 |
| 8 | **8. BIBLIOGRAPHY** | 50 |

**1.INTRODUCTION**

**1.INTRODUCTION**

Cafe Management System (CMS) helps the user to view the bill details of their customer purchased items. In this system the user knows the place order, view bill and order place details. By using this CMS, the admin need not worry about the sales, profit of their investment without hesitation that is in previous approach the admin wants to know the sales and lose details he/she has to calculate their own which can’t give the accurate sales and lose details efficiently. In this project an attempt is made to design a computer system for the CAFÉ that makes the management of recording user details, and billing much easier.

**1.1ABSTRACT**

In this project the Cafe Management System, the objective of this application is to view the bill. It makes the users work simple, efficient and save our time. The user first sign up for set up an account and they should give the details of their name, Email, mobile number, address, password, security question, answer. Then enter to login page they should give the details of their Email and password then only they go to home page. This Café Shop Management the admin need not worry about the sales, profit of their investment without hesitation that is in previous approach the admin wants to know the sales details he/she has to calculate their own which can’t give the accurate sales details efficiently. Our proposed system will overcome the drawback of the existing system. This Café Management System is fast compared to existing system and it is an efficient.

**2. SYSTEM ANALYSIS**

**2.SYSTEM ANALYSIS**

System analysis is for finding out what happens in existing system deciding on what changes and new features are required and defining exactly what the proposed system must be. The process of system analysis is largely concerned with determining, developing and agreeing to the user’s requirements. I t provides prime opportunity to communicate well with the user and conceive a joint understanding of what a system should be doing together with a view of relative importance of the system facilities using interactive techniques.

**2.1EXISTING SYSTEM**

Presently, most of the functions in the cafes are done manually. The admin also has to calculate the total income of a day. In the existing system admin can view the bill and control the user details only. This does not provide the hole sales details. By using previous approach, the admin can only view separate sales detail only and all at one time.

**Disadvantages**

* Search sales result separately for every sale.
* Hard to find the sales and profit details.
* Chances of error.

Focusing on the drawbacks and inadequacies of the existing system, the new system is designed which could well replace the existing system.

**2.2PROPOSED SYSTEM**

The main of proposed system is to view the sales and profit. The admin and user both can view the all sales and profit details. By utilizing this feature the admin can easily identify the profit and previous month or year sales details easily.

**Advantages**

* Viewing of all sales details at any time.
* The admin need not worry about the calculate the sales or profit.
* User friendly.

**2.3MODULE DESCRIPTION**

* + Signup & login.
  + User function.
  + Change password & security question.
  + Manage Category.
  + Verify Users.
  + Manipulate product.

**Signup & login**

Admin plays the main role of this project and admin login into the system will have additional functionalities when compared to normal user. Admin will monitor overall process in this system. Product details are added and updated by the administrator. Third party cannot login into system. Here user details and product related changes are maintained in the secure way.

**User Function**

In this module, each user will have their own email and password to login into system. This login process is created by the users but user login was handled by admin. The user has to login and place order, view bill and order place details of the purchased items.

**Change password & security question**

In this module, the user and admin both are use this option. They want to change their password or security question they will change it using their old password and old security question. All this changed data are automatically save in database.

**Manage Category**

In this module, the admin only handles this option not for user. This will be adding the new category products for their shop food menu. And also delete the new category by click the row of the category product.

**Verify Users**

In this module, the admin only handles this option not for user. This will be use for monitor the user process such as who are use the system. And also search one particular user by their Email id.

**Manipulate product**

In this module, the admin only handles this option not for user. This will be used for add new product for particular category and insert, delete and update product by using their category and name of the product.

**3. SYSTEM SPECIFICATION**.

**3.1 HARDWARE REQUIREMENT**

* System : Lenovo.
* Hard Disk : 400 GB.
* Monitor : LCD.
* Mouse : Laptop Touch pad.
* Ram : 4 Gb.

**3.2 SOFTWARE REQUIREMENTS**

* Operating system : Windows 10.
* Coding Language : VB.NET
* Tool : Visual Studio
* Data Base : Oracle 20c

**3.3 SOFTWARE DESCRIPTION**

**THE .NET FRAMEWORK:**

The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet.

**OBJECTIVES OF .NET FRAMEWORK:**

* To provide a consistent object-oriented programming environment whether object codes is stored and executed locally on Internet-distributed, or executed remotely.
* To provide a code-execution environment to minimizes software deployment and guarantees safe execution of code.
* Eliminates the performance problems.

There are different types of application, such as Windows-based applications and Web-based applications. To make communication on distributed environment to ensure that code be accessed by the .NET Framework can integrate with any other code.

**COMPONENTS OF. NET FREAMWORK:**

**THE COMMON LANGUAGE RUNTIME (CLR):**

The common language runtime is the foundation of the .NET Framework. It manages code at execution time, providing important services such as memory management, thread management, and remoting and also ensures more security and robustness. The concept of code management is a fundamental principle of the runtime. Code that targets the runtime is known as managed code, while code that does not target the runtime is known as unmanaged code.

**THE .NET FRAME WORK CLASS LIBRARY*:***

It is a comprehensive, object-oriented collection of reusable types used to develop applications ranging from traditional command-line or graphical user interface (GUI) **NET FRAMEWORK:** applications to applications based on the latest innovations provided by VB.NET, such as Web Forms and XML Web services.

The .NET Framework can be hosted by unmanaged components that load the common language runtime into their processes and initiate the execution of managed code, thereby creating a software environment that can exploit both managed and unmanaged features. The .NET Framework not only provides several runtime hosts, but also supports the development of third-party runtime hosts.

Internet Explorer is an example of an unmanaged application that hosts the runtime (in the form of a MIME type extension). Using Internet Explorer to host the runtime to enables embeds managed components or Windows Forms controls in HTML documents.

**FEATURES OF THE COMMON LANGUAGE RUNTIME:**

The common language runtime manages memory; thread execution, code execution, code safety verification, compilation, and other system services these are all run on CLR.

* Security.
* Robustness.
* Productivity.
* Performance.

**Security:**

The runtime enforces code access security. The security features of the runtime thus enable legitimate Internet-deployed software to be exceptionally feature rich. With regards to security, managed components are awarded varying degrees of trust, depending on a number of factors that include their origin to perform file-access operations, registry-access operations, or other sensitive functions.

**ROBUSTNESS:**

The runtime also enforces code robustness by implementing a strict type- and code-verification infrastructure called the common type system (CTS). The CTS ensures that all managed code is self-describing. The managed environment of the runtime eliminates many common software issues.

**PRODUCTIVITY:**

The runtime also accelerates developer productivity. For example, programmers can write applications in their development language of choice, yet take full advantage of the runtime, the class library, and components written in other languages by other developers.

**PERFORMANCE:**

The runtime is designed to enhance performance. Although the common language runtime provides many standard runtime services, managed code is never interpreted. A feature called just-in-time (JIT) compiling enables all managed code to run in the native machine language of the system on which it is executing. Finally, the runtime can be hosted by high-performance, server-side applications, such as Microsoft® SQL Server™ and Internet Information Services (IIS).

**FEATURES OF VB.NET**

##### VB.NET

VB.NET stands for Visual Basic.NET, and it is a computer programming language developed by Microsoft. It was first released in 2002 to replace Visual Basic 6. VB.NET is an object-oriented programming language. This means that it supports the features of object-oriented programming which include encapsulation, polymorphism, abstraction, and inheritance.

Visual Basic .NET (VB.NET) is an object-oriented computer programming language implemented on the .NET Framework. Although it is an evolution of classic Visual Basic language, it is not backwards-compatible with VB6, and any code written in the old version does not compile under VB.NET.

Like all other .NET languages, VB.NET has complete support for object-oriented concepts. Everything in VB.NET is an object, including all of the primitive types (Short, Integer, Long, String, Boolean, etc.) and user-defined types, events, and even assemblies. All objects inherit from the base class Object.

VB.NET is implemented by Microsoft's .NET framework. Therefore, it has full access to all the libraries in the .Net Framework. It's also possible to run VB.NET programs on Mono, the open-source alternative to .NET, not only under Windows, but even Linux or Mac OSX.

The following reasons make VB.Net a widely used professional language −

* Modern, general purpose.
* Object oriented.
* Component oriented.
* Easy to learn.
* Structured language.
* It produces efficient programs.
* It can be compiled on a variety of computer platforms.
* Part of .Net Framework.

## Strong Programming Features VB.Net

VB.Net has numerous strong programming features that make it endearing to multitude of programmers worldwide. Let us mention some of these features –

* Boolean Conditions
* Automatic Garbage Collection
* Standard Library
* Assembly Versioning
* Properties and Events
* Delegates and Events Management
* Easy-to-use Generics
* Indexers
* Conditional Compilation
* Simple Multithreading

VB.NET comes loaded with numerous features that have made it a popular programming language amongst programmers worldwide. These features include the following;

* Automatic code formatting, XML designer, improved object browser etc.
* Garbage collection is automated.
* Support for Boolean conditions for decision making.
* Simple multithreading, allowing your apps to deal with multiple tasks simultaneously.
* Simple generics.
* A standard library.
* Events management.
* References. You should reference an external object that is to be used in a VB.NET application.
* Attributes, which are tags for providing additional information regarding elements that have been defined within a program.
* Windows Forms- you can inherit your form from an already existing form.
* VB.NET is not case sensitive like other languages such as C++ and Java.
* It is an object-oriented programming language. It treats everything as an object

## Modules in VB.Net

A VB.Net program consists of the following modules:

* Namespace declaration
* One or more procedures
* A class or module
* Variables
* The Main procedure
* Comments
* Statements & Expressions

## Advantages of VB.NET

The following are the pros/benefits you will enjoy for coding in VB.NET:

* Your code will be formatted automatically.
* You will use object-oriented constructs to create an enterprise-class code.
* You can create web applications with modern features like performance counters, event logs, and file system.
* You can create your web forms with much ease through the visual form designer. You will also enjoy drag and drop capability to replace any elements that you may need.
* You can connect your applications to other applications created in languages that run on the .NET framework.
* You will enjoy features like docking, automatic control anchoring, and in-place menu editor all good for developing web applications.

**DATA ACCESS WITH ADO.NET:**

As you develop applications using ADO.NET, you will have different requirements for working with data. You might never need to directly edit an XML file containing data - but it is very useful to understand the data architecture in ADO.NET.

ADO.NET offers several advantages over previous versions of ADO:

* Interoperability
* Maintainability
* Programmability
* Performance Scalability

**INTEROPERABILITY:**

ADO.NET applications can take advantage of the flexibility and broad acceptance of XML. Because XML is the format for transmitting datasets across the network, any component that can read the XML format can process data. The receiving component need not be an ADO.NET component.

The transmitting component can simply transmit the dataset to its destination without regard to how the receiving component is implemented. The destination component might be a Visual Studio application or any other application implemented with any tool whatsoever.

The only requirement is that the receiving component be able to read XML. SO, XML was designed with exactly this kind of interoperability in mind.

**MAINTAINABILITY:**

In the life of a deployed system, modest changes are possible, but substantial, Architectural changes are rarely attempted because they are so difficult. As the performance load on a deployed application server grows, system resources can become scarce and response time or throughput can suffer. Faced with this problem, software architects can choose to divide the server's business-logic processing and user-interface processing onto separate tiers on separate machines. In effect, the application server tier is replaced with two tiers, alleviating the shortage of system resources. If the original application is implemented in ADO.NET using datasets, this transformation is made easier.

ADO.NET data components in Visual Studio encapsulate data access functionality in various ways that help you program more quickly and with fewer mistakes.

**PERFORMANCE:**

ADO.NET datasets offer performance advantages over ADO disconnected record sets. In ADO.NET data-type conversion is not necessary.

**SCALABILITY:**

ADO.NET accommodates scalability by encouraging programmers to conserve limited resources. Any ADO.NET application employs disconnected access to data; it does not retain database locks or active database connections for long durations.

**VISUAL STUDIO .NET:**

Visual Studio .NET is a complete set of development tools for building ASP Web applications, XML Web services, desktop applications, and mobile applications. In addition to building high-performing desktop applications, you can use Visual Studio's powerful component-based development tools and other technologies to simplify team-based design, development, and deployment of Enterprise solutions. Visual Basic .NET, Visual C++ .NET, and Visual C# .NET all use the same integrated development environment (IDE), which allows them to share tools and facilitates in the creation of mixed-language solutions.

In addition, these languages leverage the functionality of the .NET Framework and simplify the development of ASP Web applications and XML Web services.

Visual Studio supports the .NET Framework, which provides a common language runtime and unified programming classes; ASP.NET uses these components to create ASP Web applications and XML Web services. Also, it includes MSDN Library, which contains all the documentation for these development tools.

**ORACLE Database 20c:**

[Larry Ellison](https://en.wikipedia.org/wiki/Larry_Ellison) and his two friends and former co-workers, [Bob Miner](https://en.wikipedia.org/wiki/Bob_Miner) and [Ed Oates](https://en.wikipedia.org/wiki/Ed_Oates), started a consultancy called Software Development Laboratories (SDL) in 1977. SDL developed the original version of the Oracle software. The name *Oracle* comes from the code-name of a [CIA](https://en.wikipedia.org/wiki/Central_Intelligence_Agency)-funded project Ellison had worked on while formerly employed by [Ampex](https://en.wikipedia.org/wiki/Ampex).

Oracle Database (commonly referred to as Oracle DBMS, Oracle Autonomous Database, or simply as Oracle) is a [multi-model](https://en.wikipedia.org/wiki/Multi-model_database)[[4]](https://en.wikipedia.org/wiki/Oracle_Database#cite_note-4) [database management system](https://en.wikipedia.org/wiki/Database_management_system) produced and marketed by Corporation. It is a database commonly used for running [online transaction processing](https://en.wikipedia.org/wiki/Online_transaction_processing) (OLTP), [data warehousing](https://en.wikipedia.org/wiki/Data_warehouse) (DW) and mixed (OLTP & DW) database workloads. Oracle Database is available by several service providers [on-prem](https://en.wikipedia.org/wiki/On-premises_software), [on-cloud](https://en.wikipedia.org/wiki/Cloud_computing), or as hybrid cloud installation. It may be run on third party servers as well as on Oracle hardware ([Exadata](https://en.wikipedia.org/wiki/Oracle_Exadata) on-prem, on [Oracle Cloud](https://en.wikipedia.org/wiki/Oracle_Cloud) or at Cloud at Customer ). Intrinsic awareness of multimedia data will allow specialized functions to be performed on them

1. SQL Macros
2. Enhanced SQL operations including SET operation.
3. High performance SQL JSON data type (stored in binary format) and autonomous JSON database option in the cloud
4. Analytic window function chaining
5. New CHEKSUM function
6. Enhanced machine learning and AI capabilities for both data scientist and non-ML users.

## ****1.SQL Macros****

## As a new re-usability feature, SQL macro allows shared logic encapsulation into reusable, parameterized macros that can be used in other SQL statements. To make use of this feature, **create a functionwith theSQL\_MACRO(SCALAR) or SQL\_MACRO clauses in its definition**. This makes the function behave as a macro just like any other “macro”. The former creates a scalar macro (SCALAR type) that can be called in SELECT, WHERE, GROUP BY, HAVING, and ORDER BY clauses; and the latter creates a table macro (TABLE type) that can ****only**** be called in the FROM clause of a query. However, it can return a VARCHAR2 or CLOB value only. SQL Macros always run as invoker right’s functions and need INHERIT PRIVILEGES to be granted to it by its owner. Virtual columns, MVs, function-based indexes, and editioning views cannot reference a SQL\_MACRO function.

## This adds a strong security layer for these macro-defined functions due to INHERIT privileges and invoker rights combination. SQL macros also enhance code quality. And can be adopted as enterprise standards for SQL/PLSQL. Secondly, SQL\_MACRO based functions avoid context-switching between SQL and PL/SQL. Thirdly, SQL\_MACRO exposes SQL code transparent to SQL optimizer thereby resulting in efficient execution. Fourthly, SQL\_MACRO supports cross-DB compatibility by way of mimicking code in a different database to be written in equivalent Oracle SQL code but giving the ability to use it in the same manner as in the other DB.

## ****2. Enhanced SQL operations including SET operations****

* Cross-data type querying including NoSQL, JSON, XML, Graph, Spatial, and Files as well as transactions across all of these data types.
* MINUS [ALL], INTERSECT [ALL], and **a new construct EXCEPT [ALL] that is the logical equivalent of MINUS [ALL]** are introduced. This way complete ANSI SQL compliance is achieved that is much needed for migration and portability from other DB to Oracle. The below examples illustrate the use of EXCEPT and EXCEPT ALL.
* The INSERT ALL returns ALL rows in the first set that are there in the second set including duplicates across the columns selected.

## ****3. High-performance SQL JSON data type (stored in binary format) and Autonomous JSON Database option in the Cloud****

A new data type using the keyword JSON can be used in DDL statements such as CREATE TABLE or DMLs for JSON-enabled data. This is optimal for storing large JSON documents in-db and is optimized for query and DML involving JSON-type documents as well as for indexing. The JSON data type takes less storage than CLOBs and also resolves character-set differences (no character-set conversions are done). A number of JSON based functions in 20c allow declarative use with JSON column values to update, delete, or query such JSON column elements. JSON\_TRANSFORM can be used in an UPDATE statement to set and remove JSON object elements.

## ****4. Analytic window function chaining****

Oracle 20c enables window chaining when using analytic functions with the OVER clause. The window clause is specified as part of table expression and windowing functions use the window name thus specified.

## ****5. New CHECKSUM function****

A SQL-native CHECKSUM function in Oracle 20c now allows calculating the hash value of a SQL expression. Instead of a making a PL/SQL call as in pre-20c using DBMS\_SQLHASH.GET\_HASH (to check integrity of result sets) or such other API (such as standard hash(), dbms\_crypto etc.), the CHECKSUM can be directly called in a SQL SELECT clause thus providing an easy and efficient way of preserving data integrity. Secondly, there is **no need of any GRANTS to be given** to use this function.

## ****6.Enhanced machine learning and AI capabilities for both data scientists and non-ML****

## **These following are include:** a. New in-DB native new ML algorithms for deep learning model build and real-time scoring on new data such as XGBOOST algorithm that can be used for predictions like propensity to buy, and classification like text classification tasks; MSET-SPRT algorithm for anomaly detection. b. Custom R and Python scripts to be run in-DB to augment these algorithms and can be deployed via REST API. c. Auto-build of ML models for model selection, feature selection, and hyper-parameter tuning. d. Recommendation of models for performance.

**Features of SQL Server**

* License cost is very lower than any RDBMS Systems
* Runs under Windows all windows servers (NT/2000/2003)
* Runs under client versions Windows 95/98/XP … (to install server service, we need to install MSDE – Microsoft SQL Desktop Engine)
* Scalable to meet Enterprise level databases
* Supports data replication
* Supports Data Marts and Data Warehouses
* Provide OLAP service
* English Query tool makes data move available to casual users
* Data transmission services enable easy exchange of data
* Supports distributed transaction
* Centralized management
* Availability of Visual administration tools and wizards
* Generation and transmission of data in XML format

**Advantages of relational databases**

* Redundancy can be reduced
* Inconsistency can avoid
* Irrelevant data can be avoided
* Data can be shared
* Standards can be enforced
* Security restrictions can be applied means access levels can be defined
* Integrity can be maintained
* Conflicting requirements can be balanced

**SQL Server Database objects**

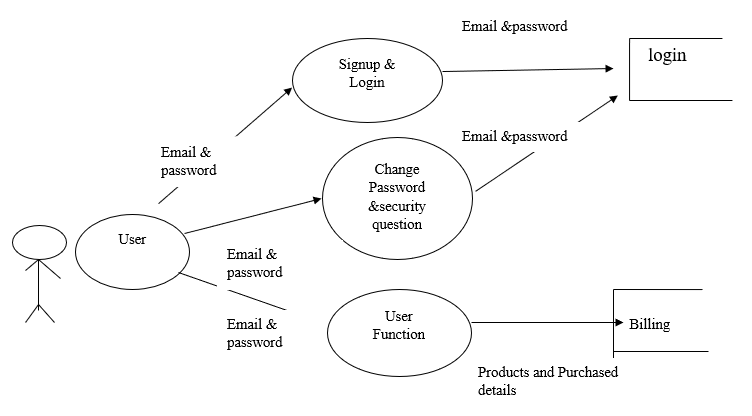
In a SQL server the following database object can be created and maintained.

* Database Users (user accounts)
* Databases
* Tables
* Views
* Constraints
* Indexes
* Triggers, Stored procedures and functions

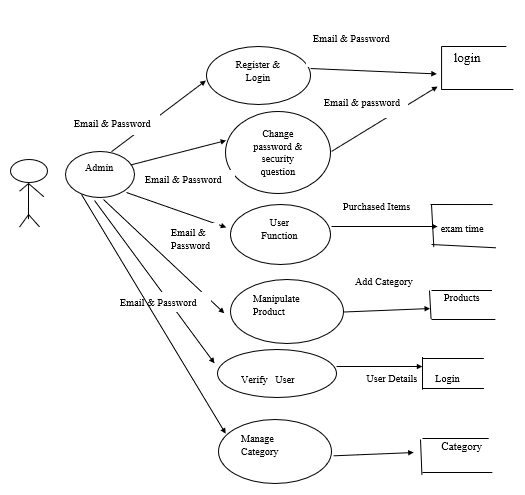
4. SYSTEM DESIGN.

**4.1DATA FLOW DIAGRAM**

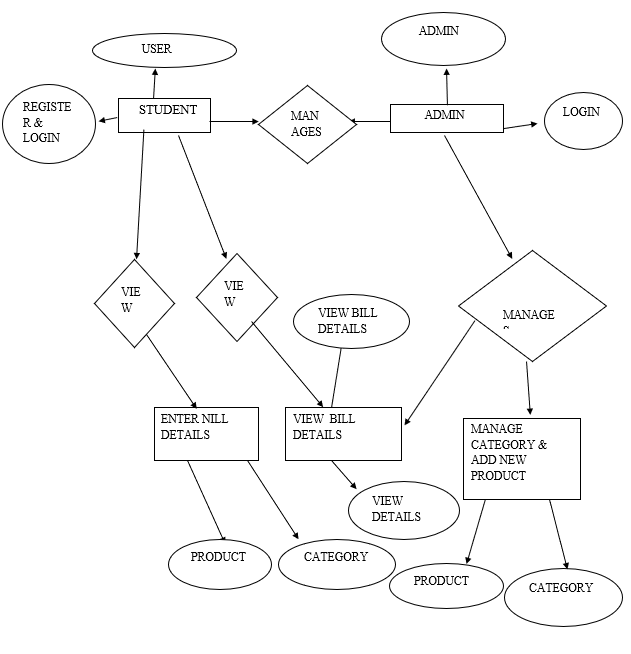
**USER**



**ADMIN**

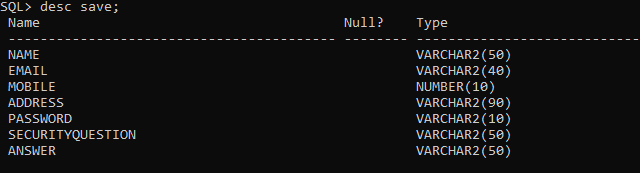


**4.2ENTITY RELATION DIAGRAM**

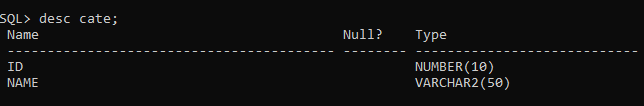
****

**4.3 TABLE STRUCTURE:**

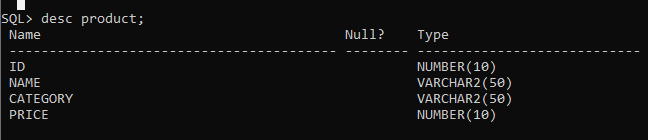
**Register & Login:**

****

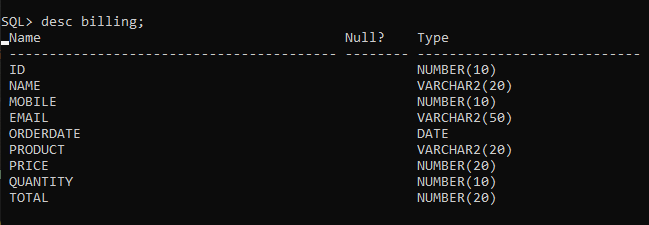
**Category :**

****

**Product:**

****

**Billing:**

****

5. PROJECT IMPLEMENTATION

**5.1 SAMPLE CODING**

**Register & Login**

Imports System

Imports System.Data

Imports System.Data.OracleClient

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

If TextBox3.Text.Contains("@") And TextBox3.Text.Contains(".") Then

If (TextBox2.Text = "" Or TextBox3.Text = "" Or TextBox4.Text = "" Or TextBox5.Text = "" Or TextBox6.Text = "" Or TextBox7.Text = "" Or TextBox8.Text = "") Then

MessageBox.Show("Please Fill all the details")

Else

Dim con As New OracleConnection

Dim cmd As New OracleCommand

Dim st As String

st = "insert into save values('" & TextBox2.Text & "','" & TextBox3.Text & "','" & TextBox4.Text & "','" & TextBox5.Text & "','" & TextBox6.Text & "','" & TextBox7.Text & "','" & TextBox8.Text & "')"

con = New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

cmd = New OracleCommand(st, con)

cmd.ExecuteNonQuery()

MessageBox.Show("Registered Successfully")

End If

Else

MessageBox.Show("Please Enter Vaild Email")

End If

End Sub

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click

If TextBox1.Text.Contains("@") And TextBox1.Text.Contains(".") Then

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

Dim cmd As New OracleCommand("select \* from save where email='" & TextBox1.Text & "'and password='" & TextBox2.Text & "'", con)

Dim da As New OracleDataAdapter(cmd)

Dim pass As String = cmd.ExecuteScalar()

Dim dt As DataTable = New DataTable()

da.Fill(dt)

If (TextBox1.Text = "admin@gmail.com") Then

MessageBox.Show("Admin Login Success", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

home.Show()

Me.Hide()

ElseIf (dt.Rows.Count > 0) Then

MessageBox.Show("Login Success", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

homeus.Show()

Me.Hide()

Else

MessageBox.Show("Error", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

End If

Else

MessageBox.Show("Please enter valid Email address", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

End If

End Sub

**Change Password & security Question:**

**Change Password:**

Imports System

Imports System.Data

Imports System.Data.OracleClient

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles Button2.Click

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

Dim cmd As New OracleCommand("update save set password='" & TextBox3.Text & "'where email='" & TextBox4.Text & "'", con)

cmd.ExecuteNonQuery()

MessageBox.Show("Updated Successfully", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

End Sub

<Obsolete>

Private Sub TextBox1\_TextChanged(sender As Object, e As EventArgs) Handles TextBox1.TextChanged

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

Dim cmd As New OracleCommand("select password from save where password='" & TextBox1.Text & "'", con)

Dim pass As String = cmd.ExecuteScalar()

If (pass = TextBox1.Text) Then

Label5.Text = "OK"

Label5.ForeColor = Color.Lime

Button2.Enabled = True

Else

Label5.Text = "WRONG"

Label5.ForeColor = Color.Red

Button2.Enabled = False

End If

con.Close()

End Sub

**Change Security Qusetion:**

Imports System

Imports System.Data

Imports System.Data.OracleClient

Private Sub TextBox4\_TextChanged(sender As Object, e As EventArgs) Handles TextBox4.TextChanged

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

Dim cmd As New OracleCommand("select password from save where password='" & TextBox4.Text & "'", con)

Dim pass As String = cmd.ExecuteScalar()

If (pass = TextBox4.Text) Then

Label7.Text = "OK"

Label7.ForeColor = Color.Lime

Button2.Enabled = True

Else

Label7.Text = "WRONG"

Label7.ForeColor = Color.Red

Button2.Enabled = False

End If

con.Close()

End Sub

<Obsolete>

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles Button2.Click

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

Dim cmd As New OracleCommand("update save set securityquestion='" & TextBox2.Text & "',answer='" & TextBox3.Text & "'where password='" & TextBox4.Text & "'", con)

cmd.ExecuteNonQuery()

MessageBox.Show("Updated Successfully", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

End Sub

**Manage Category:**

Imports System

Imports System.Data

Imports System.Data.OracleClient

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles Button2.Click

If (TextBox1.Text = "" Or TextBox2.Text = "") Then

MessageBox.Show("Please Fill all the details", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

Else

Dim con As New OracleConnection

Dim cmd As New OracleCommand

Dim st As String

st = "insert into cate values('" & TextBox2.Text & "','" & TextBox1.Text & "')"

con = New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

cmd = New OracleCommand(st, con)

cmd.ExecuteNonQuery()

MessageBox.Show("Category Added Successfully", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

End If

BindGrid()

End Sub

<Obsolete>

Private Sub DataGridView1\_CellClick(sender As Object, e As DataGridViewCellEventArgs) Handles DataGridView1.CellClick

Dim confirm = MessageBox.Show("Are you sure to delete?", "Delete", MessageBoxButtons.YesNo)

If confirm = DialogResult.Yes Then

Dim sri As DataGridViewRow

sri = DataGridView1.Rows(e.RowIndex)

Dim pid As Integer = sri.Cells(0).Value

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

Dim cmd As New OracleCommand("delete cate where id='" & pid & "'", con)

cmd.ExecuteNonQuery()

BindGrid()

End If

End Sub

**Verify User:**

Imports System

Imports System.Data

Imports System.Data.OracleClient

Private Sub GridBind()

Using con As OracleConnection = New OracleConnection("data source=xe;user id=system;password=2020it17")

Using cmd As OracleCommand = New OracleCommand("select \* from save", con)

Using da As New OracleDataAdapter

da.SelectCommand = cmd

Using dt As New DataTable

da.Fill(dt)

DataGridView1.DataSource = dt

End Using

End Using

End Using

End Using

End Sub

<Obsolete>

Private Sub TextBox1\_TextChanged(sender As Object, e As EventArgs) Handles TextBox1.TextChanged

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

If con.State = ConnectionState.Closed Then

con.Open()

End If

Dim cmd As New OracleCommand("select \* from save where email like '%" + TextBox1.Text + "%'", con)

Dim da As New OracleDataAdapter

Dim dt As New DataTable

da.SelectCommand = cmd

dt.Clear()

da.Fill(dt)

DataGridView1.DataSource = dt

DataGridView1.Columns(0).HeaderText = "name"

DataGridView1.Columns(1).HeaderText = "email"

DataGridView1.Columns(2).HeaderText = "mobile"

DataGridView1.Columns(3).HeaderText = "address"

DataGridView1.Columns(4).HeaderText = "password"

DataGridView1.Columns(5).HeaderText = "securityquestion"

DataGridView1.Columns(6).HeaderText = "answer"

con.Close()

End Sub

**Add New Product:**

Imports System

Imports System.Data

Imports System.Data.OracleClient

If (TextBox1.Text = "" Or TextBox2.Text = "" Or TextBox3.Text = "" Or ComboBox1.Text = "") Then

MessageBox.Show("Please Fill all the details", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

Else

Dim con As New OracleConnection

Dim cmd As New OracleCommand

Dim st As String

st = "insert into product values('" & TextBox2.Text & "','" & TextBox1.Text & "','" & ComboBox1.Text & "','" & TextBox3.Text & "')"

con = New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

cmd = New OracleCommand(st, con)

cmd.ExecuteNonQuery()

MessageBox.Show("Product Added Successfully", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

End If

End Sub

<Obsolete>

Private Sub addnewproduct\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

Dim cmd As New OracleCommand("select \* from cate", con)

Dim da As New OracleDataAdapter(cmd)

Dim dt As New DataTable()

da.Fill(dt)

ComboBox1.DataSource = dt

ComboBox1.DisplayMember = "name"

ComboBox1.ValueMember = "id"

End Sub

**View, Edit & Delete Product:**

Imports System

Imports System.Data

Imports System.Data.OracleClient

Private Sub view\_edit\_deleteproduct\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

Dim cmd As New OracleCommand("select \* from cate", con)

Dim da As New OracleDataAdapter(cmd)

Dim dt As New DataTable()

da.Fill(dt)

ComboBox1.DataSource = dt

ComboBox1.DisplayMember = "name"

ComboBox1.ValueMember = "id"

DataBind()

End Sub

<Obsolete>

Private Sub DataBind()

Using con As OracleConnection = New OracleConnection("data source=xe;user id=system;password=2020it17")

Using cmd As OracleCommand = New OracleCommand("select \* from product", con)

Using da As New OracleDataAdapter

da.SelectCommand = cmd

Using dt As New DataTable

da.Fill(dt)

DataGridView1.DataSource = dt

End Using

End Using

End Using

End Using

End Sub

Private Sub DataGridView1\_CellClick(sender As Object, e As DataGridViewCellEventArgs) Handles DataGridView1.CellClick

Dim segr As DataGridViewRow

segr = DataGridView1.Rows(e.RowIndex)

Label6.Text = segr.Cells(0).Value

TextBox1.Text = segr.Cells(1).Value

ComboBox1.Text = segr.Cells(2).Value

TextBox3.Text = segr.Cells(3).Value

End Sub

<Obsolete>

Private Sub Button2\_Click(sender As Object, e As EventArgs) Handles Button2.Click

If (TextBox1.Text = "" Or TextBox3.Text = "" Or ComboBox1.Text = "" Or Label6.Text = "") Then

MessageBox.Show("Please fill all the details", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

Else

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

Dim cmd As New OracleCommand("update product set name='" & TextBox1.Text & "',category='" & ComboBox1.Text & "',price='" & TextBox3.Text & "'where id='" & Label6.Text & "'", con)

cmd.ExecuteNonQuery()

MessageBox.Show("Product Updated Successfully", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

End If

DataBind()

End Sub

<Obsolete>

Private Sub Button3\_Click(sender As Object, e As EventArgs) Handles Button3.Click

If (TextBox1.Text = "" Or TextBox3.Text = "" Or ComboBox1.Text = "" Or Label6.Text = "") Then

MessageBox.Show("Please fill all the details", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

Else

Dim con As New OracleConnection("data source=xe;user id=system;password=2020it17")

con.Open()

Dim cmd As New OracleCommand("delete from product where id='" & Label6.Text & "'", con)

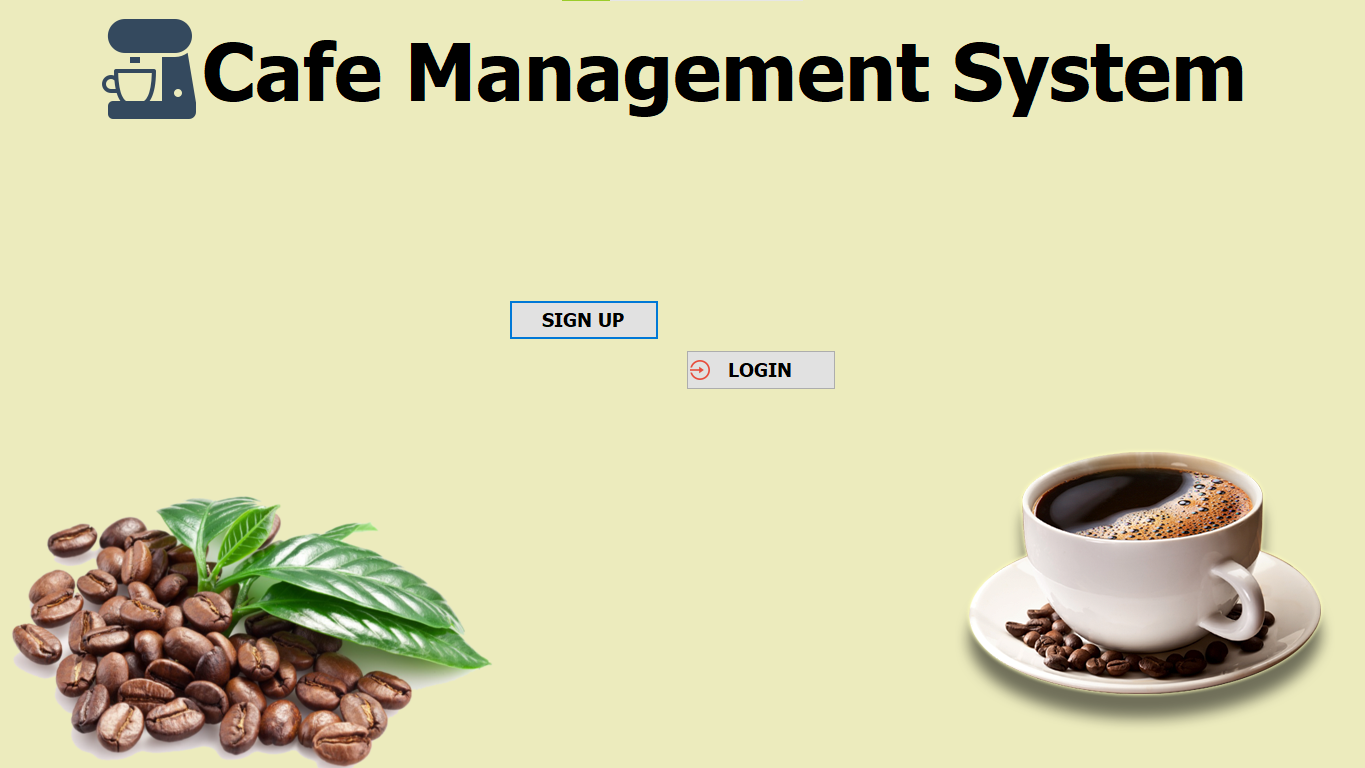
cmd.ExecuteNonQuery()

MessageBox.Show("Product Deleted Successfully", "information", MessageBoxButtons.OK, MessageBoxIcon.Information)

End If DataBind() End Sub End Class

**5.2 SAMPLE OUTPUT**

**Home Page**

****

**Signup page:**

****

**Login page:**

****

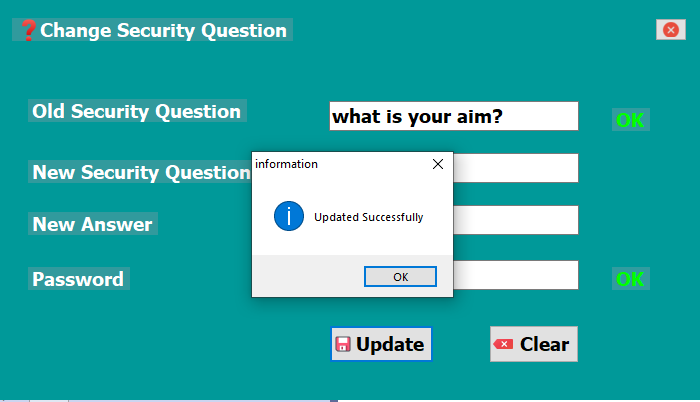
**Forget Password page:**

****

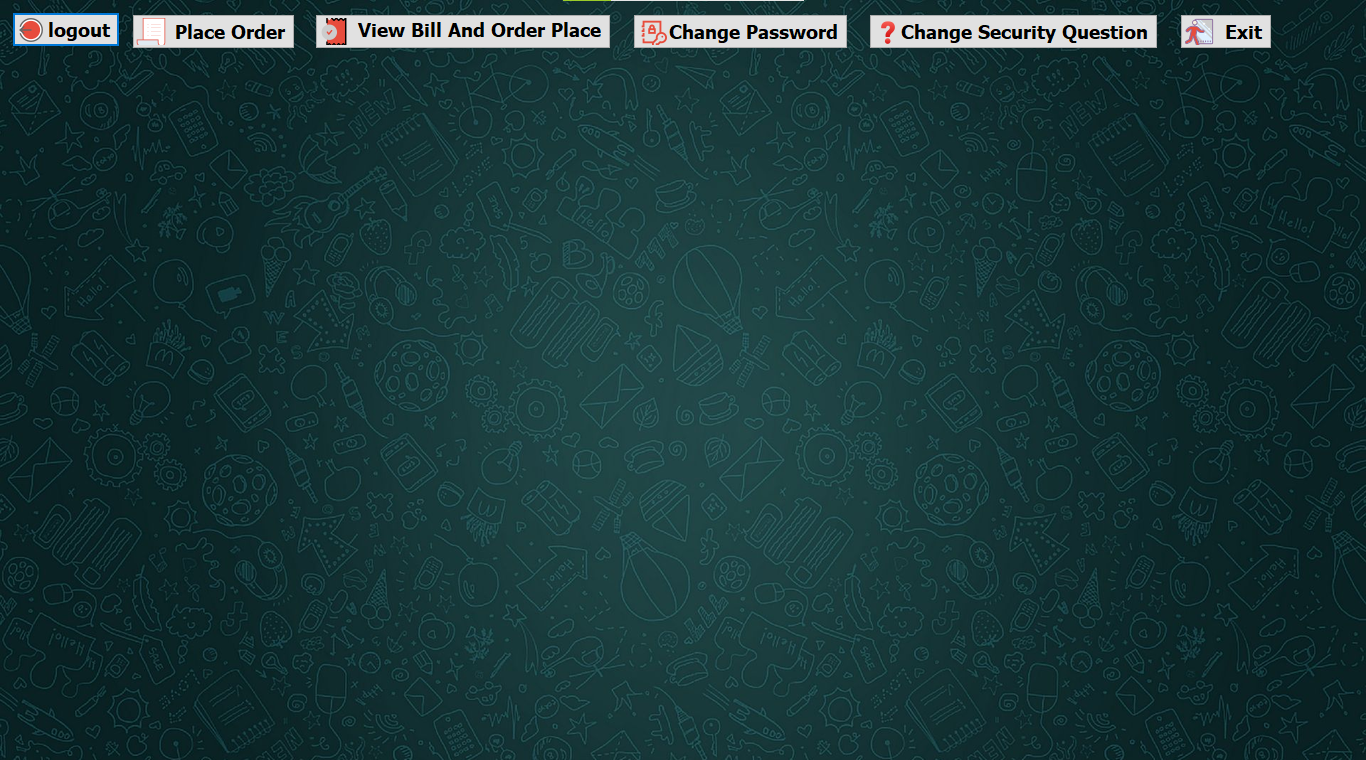
**Change Password:**

****

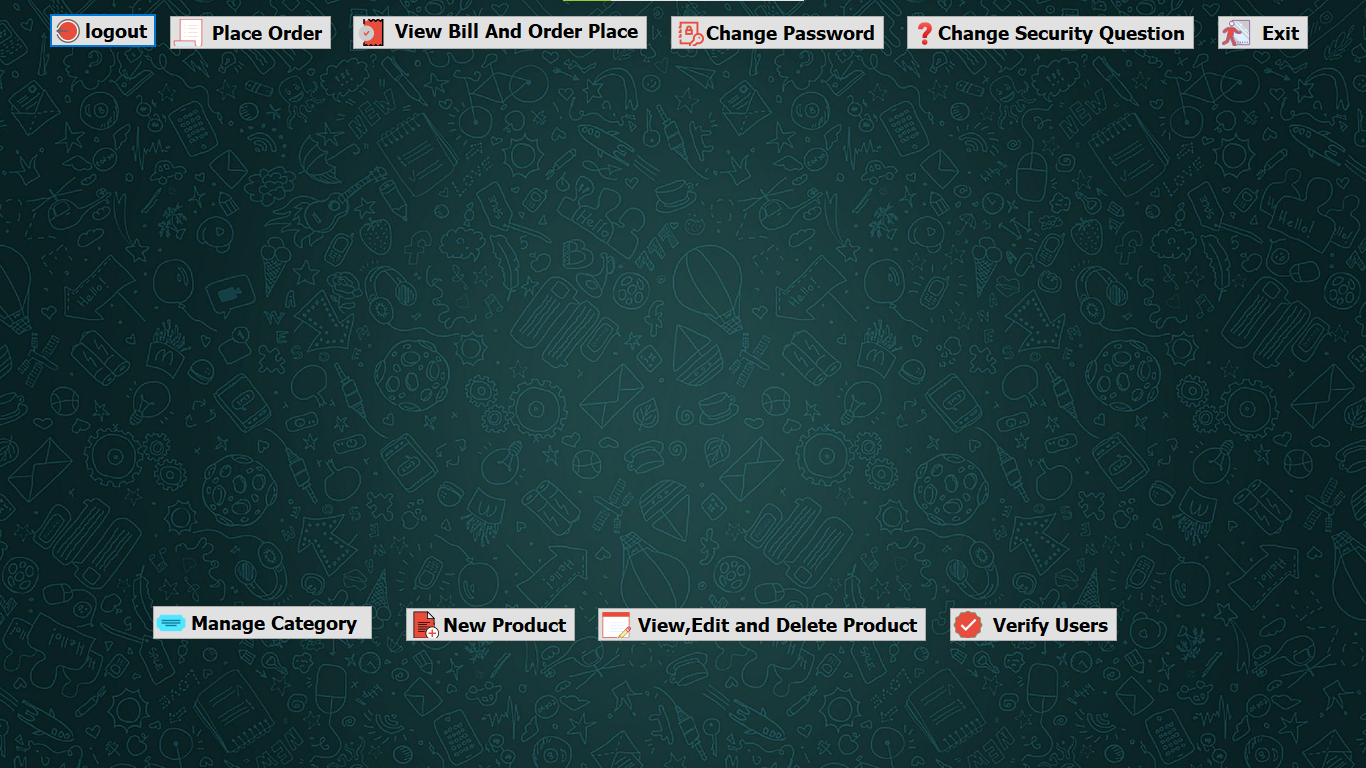
**Change Security Question:**

****

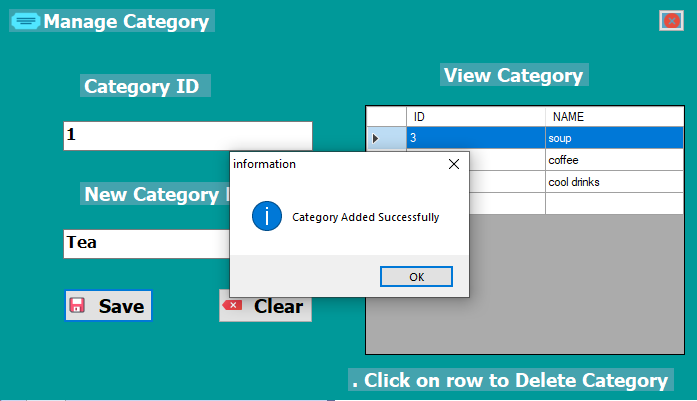
**User Home Page:**

****

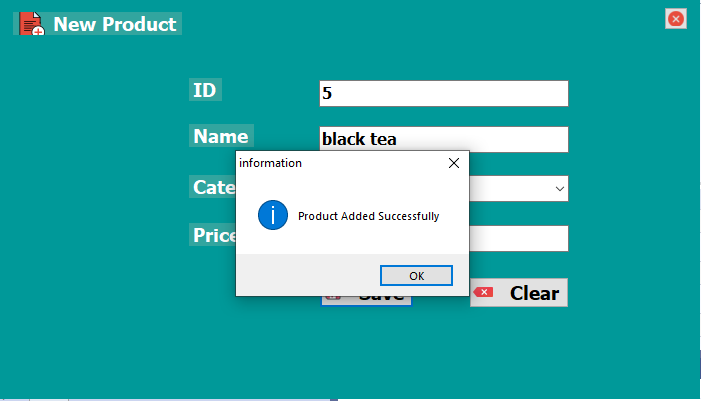
**Admin Home Page:**

****

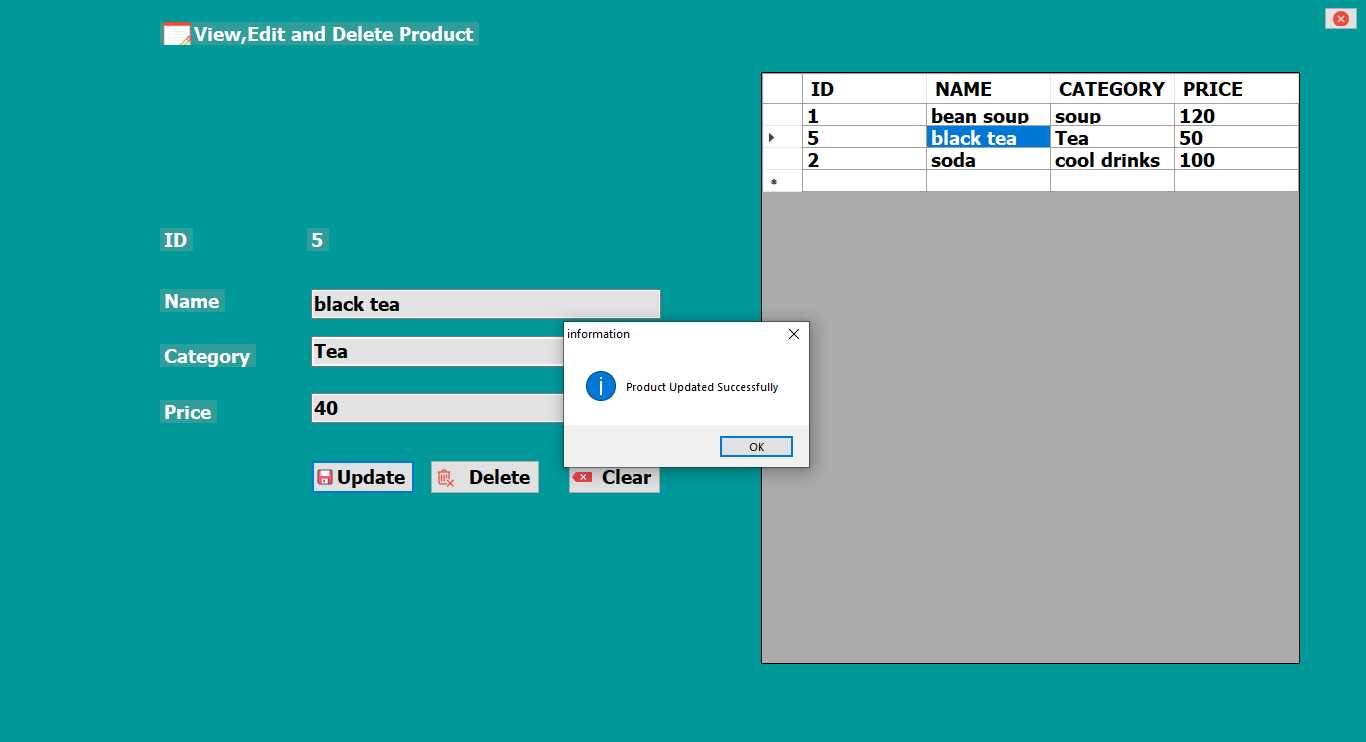
**Manage Category:**

****

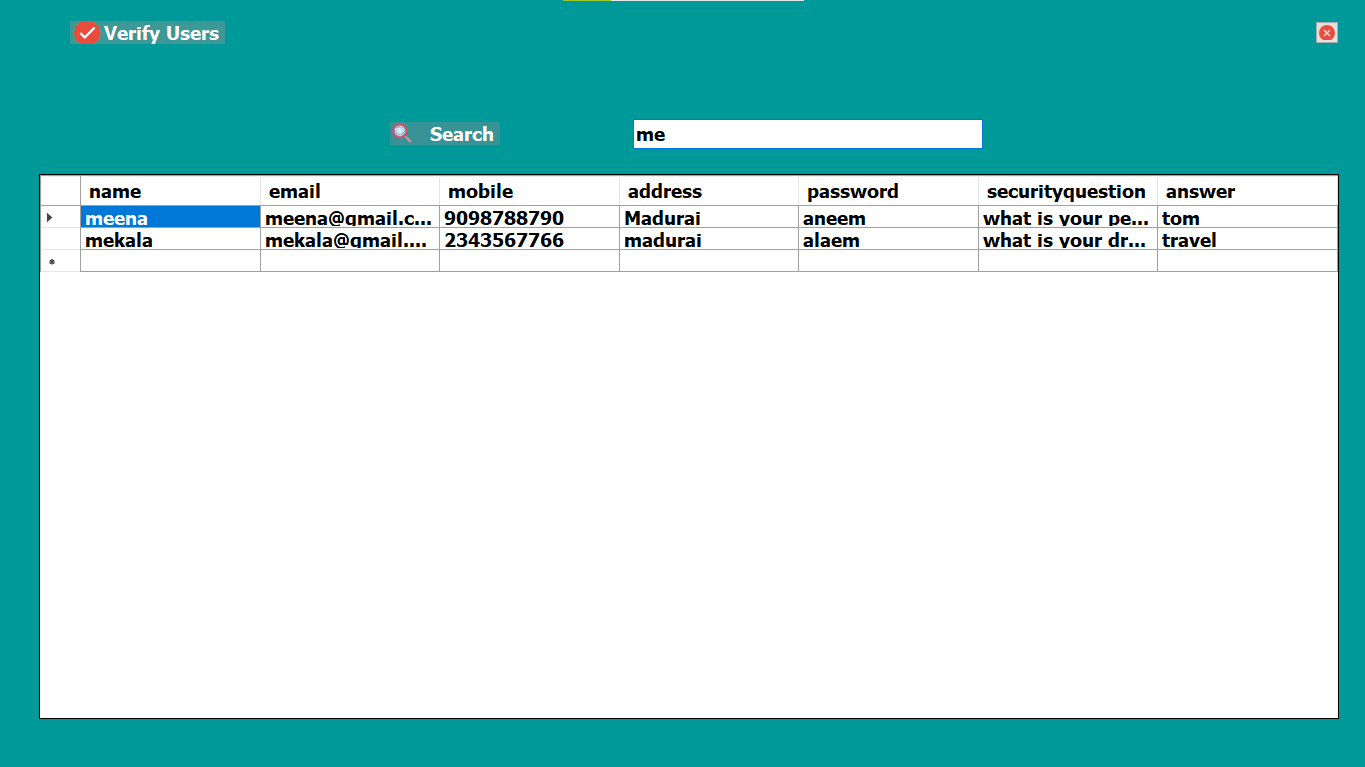
**Add New Product:**

****

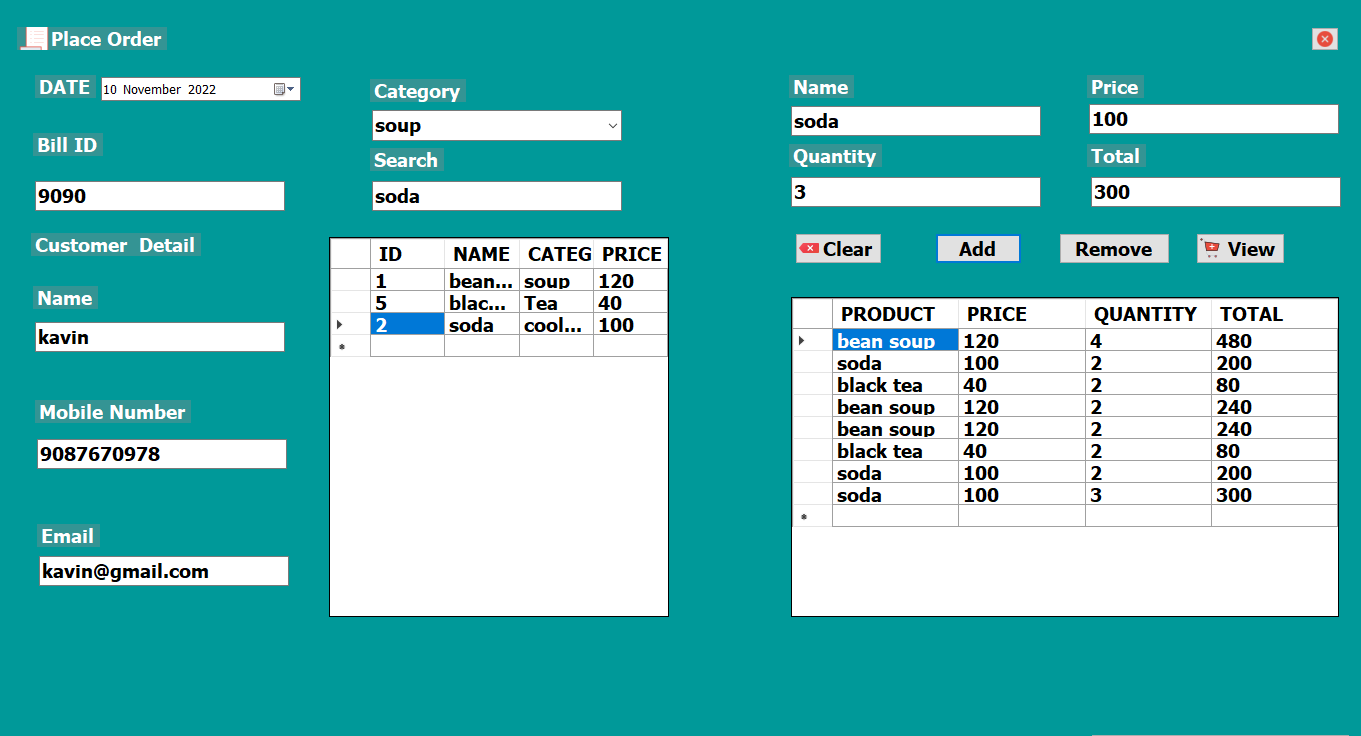
**View, Edit & Delete Product:**

****

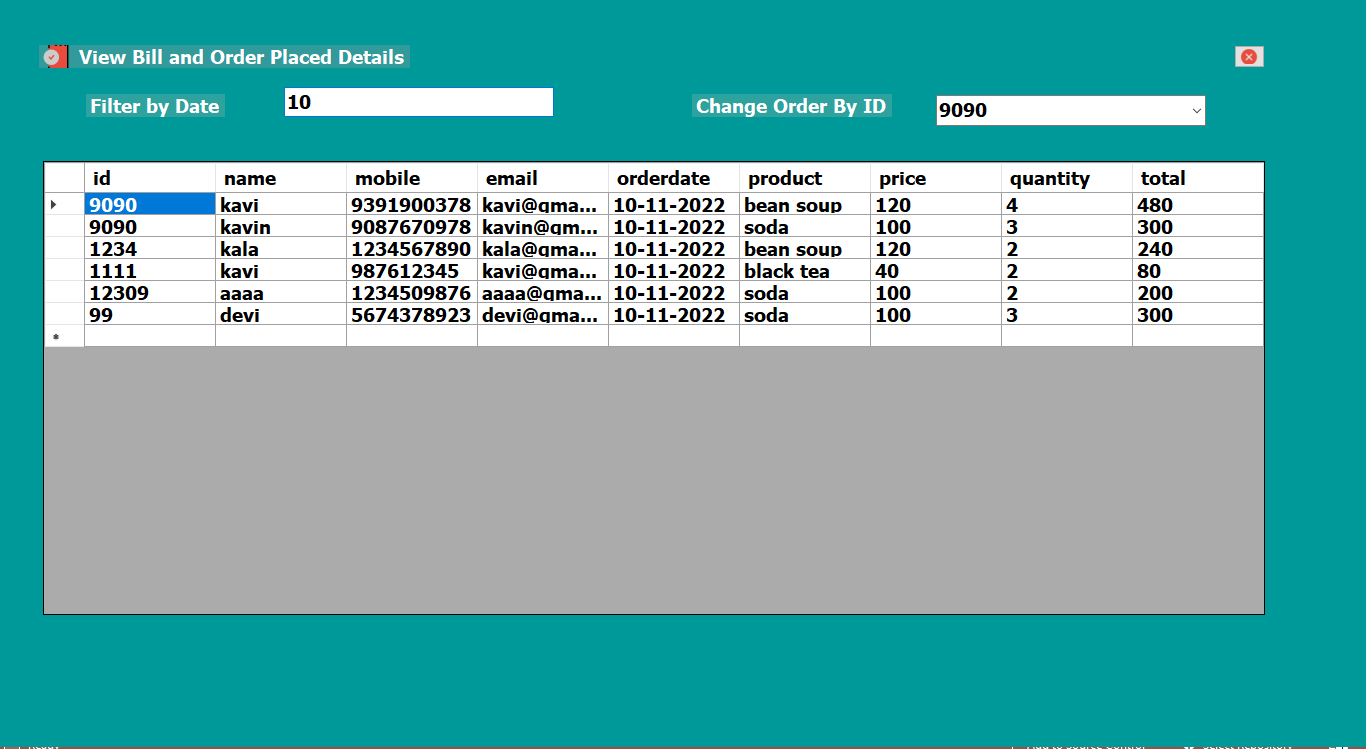
**Verify User:**

****

**Place Order:**

****

**View Order Details:**

****

6. TESTING

**6.Testing**

Testing is vital to the success of the system. System testing makes a logical assumption that if all parts of the system are correct, the goal will be successfully achieved. In the testing process we test the actual system in an organization and gather errors from the new system operates in full efficiency as stated. System testing is the stage of implementation, which is aimed to ensuring that the system works accurately and efficiently.

In the testing process we test the actual system in an organization and gather errors from the new system and take initiatives to correct the same. All the front-end and back-end connectivity are tested to be sure that the new system operates in full efficiency as stated. System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently.

The main objective of testing is to uncover errors from the system. For the uncovering process we have to give proper input data to the system. So, we should have more conscious to give input data. It is important to give correct inputs to efficient testing.

Testing is done for each module. After testing all the modules, the modules are integrated and testing of the final system is done with the test data, specially designed to show that the system will operate successfully in all its aspects conditions. Thus, the system testing is a confirmation that all is correct and an opportunity to show the user that the system works. Inadequate testing or non-testing leads to errors that may appear few months later.

This will create two problems, Time delay between the cause and appearance of the problem. The effect of the system errors on files and records within the system. The purpose of the system testing is to consider all the likely variations to which it will be suggested and push the system to its limits.

The testing process focuses on logical intervals of the software ensuring that all the statements have been tested and on the function intervals (i.e.,) conducting tests to uncover errors and ensure that defined inputs will produce actual results that agree with the required results. Testing has to be done using the two common steps Unit testing and Integration testing. In the project system testing is made as follows:

The procedure level testing is made first. By giving improper inputs, the errors occurred are noted and eliminated. This is the final step in system life cycle. Here we implement the tested error-free system into real-life environment and make necessary changes, which runs in an online fashion. Here system maintenance is done every month or year based on company policies, and is checked for errors like runtime errors, long run errors and other maintenances like table verification and reports.

**6.1 UNIT TESTING**

The first step in testing is Unit testing. Individual testing is tested to ensure that they operate correctly. Each component is tested independently, without other system components. The module interface is tested to ensure that information properly flow into and out of the program. These are tested that the module operates at boundary established to limit or restrict processing. Unit testing is normally considered as an adjunct to the coding step. After the coding has been developed, received and verified for correct syntax, unit testing begins. Here each module is tested to provide its correctness, validity and determine any missing operations and to verify whether the objectives have been met, errors are noted down and corrected immediately.

TC1: <Email=”ADMIN”, password=”admin”, expected login=”success”, original login=”success”>

TC2: <Email=”ADMIN”, password=”admin”, expected login=”success”, original login=”fail”>

TC3: <Email=”ADMIN”, password=”admin”, expected login=”success”, original login=”fail”>

TC4: <Email=”ADMIN”, password=”admin”, expected login=”success”, original login=”success”>

Unit testing is the important and major part of the project. So, errors can be rectified easily in each module and program clarity can be increased. In this project, the entire system is divided into several modules and is developed individually. Hence, unit testing is conducted to individual modules.

7. CONCLUSION

**7.1 BENEFITS**

Café Management System (CMS) is designed in order to view the customer purchased details and sales, profit details. In this project viewing of extra details of their user login details and Products in menu. Which is not given by existing system. Automatically those details are listed based on the user email id and product id.

**7.2 FUTURE ENHANCEMENT**

This system is useful for the admin and users to view the order details and user details for admin only. In future it is enhanced to provide 3D view of the sales detail for easy understanding of the admin. And various other options are also added in future for better usability of the project.

8. BIBLIOGRAPHY

MANAGES

ADMIN



EMPLOYEE

ADD STOCK

BILLING

DATA GIVES

BILLING

MANIPULATE STOCK

VIEW ORDER AND STACK DETAILS

**i) Books referred:**

* VB.NET

1. The Complete Reference by Matthew MacDonald.

* VB.NET

1. Developers Guide by Buczek

* Software Engineering

1. Roger S. Pressman.

* VB.NET

1. ASP.NET Professional by Wrox

**ii)Websites:**

[www.c-sharpcorner.com](http://www.c-sharpcorner.com)

[www.vb.net](http://www.vb.net)